



*Journées Topologie et Informatique*

**Taylor expansion,  
a round-trip between syntax and semantics.**

**Christine Tasson**

Christine.Tasson@pps.univ-paris-diderot.fr

*21 mars 2013*

- 1 **Syntactical** Taylor expansion and Resource consumption
- 2 Taylor expansion in **Semantics**
- 3 **Semantics vs Syntax** : The full abstraction question

Taylor expansion :

**From Mathematics to Computer Science.**

In Maths

$$f : \mathbb{R} \rightarrow \mathbb{R}$$

Let  $x \in \mathbb{R}$ ,

$$f(x) = \sum_n \frac{1}{n!} f^{(n)}(0) \cdot x^n$$

$x \mapsto \frac{1}{n!} f^{(n)}(0) \cdot x^n$  is  
the  $n$ -linearisation of  $f$ .

In Computer Science

$$P : \text{nat} \rightarrow \text{nat}$$

Let  $x : \text{nat}$ ,

$$P\ x = \sum_n P_n \underbrace{x \cdots x}_n$$

$P_n$ , uses exactly  $n$ -times  $x$ , is  
the  $n$ -linearisation of  $P$ .

Programs :

**Resource consumption via Taylor expansion**

# Syntactical Taylor expansion : $P_x = \sum_n P_n \underbrace{x \cdots x}_n$

$\lambda$ -calculus  $\xrightarrow{\text{Taylor Expansion}}$  Resource-calculus

**$\lambda$ -calculus :**  $M, N := x \mid \lambda x M \mid (M)N$

$$(\lambda x M)N \rightarrow M[N/x]$$

« Substitute every occurrence of  $x$  in  $M$  by  $N$ . »

**Example:**  $(\lambda x (x)x)\lambda z z \rightarrow (\lambda z z)\lambda z z \rightarrow \lambda z z$ .

**Resource calculus :**  $s, t := x \mid \lambda x s \mid \langle s \rangle [t_1 \dots t_n]$

$$\langle \lambda x s \rangle [t_1 \dots t_n] \rightarrow \partial_x(s, t_1 \dots t_n)$$

« Substitute each occurrence of  $x$  in  $s$  by one  $t_i$  if possible or reduces to 0. »

**Example:**  $\langle \lambda x \langle x \rangle [x] \rangle [\lambda z z, \lambda z z] \Rightarrow \langle \lambda z z \rangle [\lambda z z] \rightarrow \lambda z z$ .

Syntactical Taylor expansion :  $P_x = \sum_n P_n \underbrace{x \cdots x}_n$

$$\begin{array}{ccc} \text{\color{red}\lambda-calculus} & \xrightarrow{\text{Taylor Exp.}} & \text{Resource-calculus} \\ M^* & = & \sum_{t \in \mathcal{T}(M)} \frac{1}{m(t)} t \end{array}$$

**Example:**  $t \in \mathcal{T}(M)$  with  $m(t) = 2$ .

$$\begin{array}{llll} M & = & (\lambda x (x)x)\lambda z z & \rightarrow (\lambda z z)\lambda z z \rightarrow \lambda z z \\ t & = & \langle \lambda x \langle x \rangle [x] \rangle [\lambda z z, \lambda z z] & \Rightarrow \langle \lambda z z \rangle [\lambda z z] \rightarrow \lambda z z \end{array}$$

$$[(\lambda x (x)x)\lambda z z]^* = \sum_{p,q} \frac{1}{p! q!} \langle \lambda x \langle x \rangle \underbrace{[x, \dots, x]}_p \rangle \underbrace{[\lambda z z, \dots, \lambda z z]}_q$$

# Syntactical Taylor expansion and Resource Consumption

**Idea :**

$$P \ x = \sum_n P_n \underbrace{x \cdots x}_n$$

«  $P_n$  is the part of  $P$  that uses  $x$   $n$ -times. »

**Proposition :**

Let  $M \rightarrow \bullet$ . Then  $\exists! s \in \mathcal{T}(M)$  such that  $\begin{cases} s \not\rightarrow 0 \\ s \rightarrow m(s)\bullet \end{cases}$ .

«  $s$  is the version of  $M$  with the explicit resources used for computation. »

**Example:**  $M = (\lambda x (x)x)\lambda z z$  and  $t = \langle \lambda x \langle x \rangle [x] \rangle [\lambda z z, \lambda z z]$

**Conclusion :**

$$M \rightarrow^* \bullet \iff M^* = \sum_{t \in \mathcal{T}(M)} \frac{1}{m(t)} t \rightarrow^* \bullet$$



**Semantics :**

**Taylor expansion and derivatives**

**Type :**  $\sigma$   $|\sigma|$  is the set of basic values

**Data :**  $x : \sigma$   $\llbracket x \rrbracket$  part of  $\llbracket \sigma \rrbracket$

**Type :**  $\sigma \Rightarrow \tau$   $|\sigma| \times |\tau|$

**Program :**  $P : \sigma \Rightarrow \tau$   $\llbracket P \rrbracket \subseteq |\sigma| \times |\tau|$  is a relation from input to output values

**Interaction :**  $P : \sigma \Rightarrow \tau$   
 $Q : \tau \Rightarrow \psi$   $\llbracket P; Q \rrbracket = \llbracket Q \rrbracket \circ \llbracket P \rrbracket$  is the composition of relations.

In order to take into account resources, we introduce multisets.

**Type :**  $\sigma \Rightarrow \tau$   $\mathcal{M}_{\text{fin}}(|\sigma|) \times |\tau|$

**Program :**  $P : \sigma \Rightarrow \tau$   $\llbracket P \rrbracket \subset \mathcal{M}_{\text{fin}}(|\sigma|) \times |\tau|$  is a multi relation between inputs and outputs

**Interaction :**  $P : \sigma \Rightarrow \tau$   
 $Q : \tau \Rightarrow \psi$   $\llbracket P; Q \rrbracket = \llbracket Q \rrbracket \circ^! \llbracket P \rrbracket$  is the composition of multi relations.

## Proposition :

Rel is a cartesian closed category cpo-enriched, a model of various functional programming languages.

# Quantitative Semantics : Counting

In order to count the **number** of non-deterministic reductions, the **probability** to get a result,... we move to **vector spaces**.

**Type :**  $\sigma$   $\mathcal{R}^{|\sigma|}$  the set of vectors with coefficients in  $\mathcal{R}$ .

**Data :**  $x : \sigma$   $\llbracket x \rrbracket$  is a vector

**Type :**  $\sigma \multimap \tau$   $\mathcal{R}^{|\sigma| \times |\tau|}$

**Program :**  $P : \sigma \multimap \tau$   $\llbracket P \rrbracket \in \mathcal{R}^{|\sigma| \times |\tau|}$  matrix or  $\llbracket P \rrbracket : \mathcal{R}^{|\sigma|} \rightarrow \mathcal{R}^{|\tau|}$  the associated linear map.

**Interaction :**  $P : \sigma \multimap \tau$   $\llbracket P; Q \rrbracket = \llbracket Q \rrbracket \circ \llbracket P \rrbracket$  is the composition of matrix.  
 $Q : \tau \multimap \psi$

# Quantitative Semantics : Counting

**Type :**  $\sigma \Rightarrow \tau$   $\mathcal{R}^{\mathcal{M}_{\text{fin}}(|\sigma|) \times |\tau|}$

**Program :**  $P : \sigma \Rightarrow \tau$   $\llbracket P \rrbracket : \mathcal{R}^{|\sigma|} \rightarrow \mathcal{R}^{|\tau|}$  is  
an entire function.

$$\forall b \in |\tau|, \llbracket P \rrbracket(\mathbf{x})_b = \sum_{\mu \in \mathcal{M}_{\text{fin}}(|\sigma|)} \llbracket P \rrbracket_{\mu, b} \cdot \mathbf{x}^\mu$$

$$\text{with } \mathbf{x}^\mu = \prod_{a \in \text{Supp}(\mu)} x_a^{\mu(a)}$$

**Interaction :**  $P : \sigma \Rightarrow \tau$   $\llbracket P; Q \rrbracket = \llbracket Q \rrbracket \circ! \llbracket P \rrbracket$   
 $Q : \tau \Rightarrow \psi$  the composition of  
entire functions.

**Proposition :** For a suitable  $\mathcal{R}$ , we can interpret various functional programming languages.

# Quantitative Semantics and Operational semantics

**PCF<sup>or</sup>** :  $L, M, P := x \mid \lambda x M \mid (M)P \mid \text{fix}(M) \mid \underline{0} \mid \text{pred}(M) \mid \text{succ}(M)$   
 $\mid \text{if } (M = \underline{0}) \text{ then } P \text{ else } L \mid p \cdot M \mid M \text{ or } P$

$p \cdot M \xrightarrow{p} M$        $M \text{ or } P \xrightarrow{1} M$        $M \text{ or } P \xrightarrow{1} P$

**Program Analysis** :  $M$  : nat a program and  $\mathcal{R}$  a semiring.

$\mathcal{B} = \{\text{T}, \text{F}\}, \vee, \wedge, \text{F}, \text{T}, \text{F} < \text{T}$        $\llbracket M \rrbracket_n^{\mathcal{B}} = \text{T} \iff \exists M \rightarrow^* \underline{n}$ .

$\mathcal{N} = \overline{\mathbb{N}}, +, \cdot, 0, 1, \leq$        $\llbracket M \rrbracket_n^{\mathcal{N}}$  number of  $M \rightarrow^* \underline{n}$ .

$\mathcal{R} = \overline{\mathbb{R}^+}, +, \cdot, 0, 1, \leq$        $\llbracket M \rrbracket_n^{\mathcal{R}}$  probability of  $M \rightarrow^* \underline{n}$ .

$\mathcal{T} = \overline{\mathbb{N}}, \mathbf{min}, +, \infty, 0, \geq$        $\llbracket M \rrbracket_n^{\mathcal{T}}$  number of  $\beta$  and  $\text{fix}()$   
redexes induced in  $M \rightarrow^* \underline{n}$ .

## Problematics :

- If  $\mathcal{R}$  is a Field, then  $[[\sigma]] = \mathcal{R}^{|\sigma|}$  is a linear space of **infinite** dimension.
- What means :  $[[P]](\mathbf{x})_- = \sum_{\mu \in \mathcal{M}_{\text{fin}}(|\sigma|)} [[P]]_{\mu,-} \cdot \mathbf{x}^\mu$

**Solutions :** Choose  $\mathcal{R}$  and properties of  $[[\sigma]]$  for convergence

- $\mathbb{R}^+$  with usual topology Probabilistic Coherent Spaces
- $\mathbb{R}$  with discrete topology Finiteness Spaces
- $\mathbb{R}$  with usual topology Convenient Vector Spaces

## Topology :

$\sum_a x_a$  converges in  $\mathbb{R}^+$  iff the sum is absolutely convergent.

## Orthogonality :

$x, y \in \mathbb{R}^{|\sigma|}$ .

$$x \perp y \iff \sum_{a \in |\sigma|} x_a y_a \in [0, 1].$$

## Types :

$[[\sigma]]$  is a probabilistic coherent space, that is  $\begin{cases} [[\sigma]] \subseteq \mathbb{R}^{|\sigma|} \\ [[\sigma]]^{\perp\perp} = [[\sigma]] \end{cases}$

with  $[[\sigma]]^\perp = \{x \in \mathbb{R}^{|\sigma|} \mid \forall y \in [[\sigma]], x \perp y\}$

**Proposition :** Probabilistic Coherent Spaces interpret **PCF<sup>or</sup>**.



**Discrete topology :**

$\sum_a x_a$  converges in  $\mathbb{R}$  iff the sum is finite.

**Orthogonality :**

$x, y \in \mathbb{R}^{|\sigma|}$ .

$$x \perp y \iff \sum_{a \in |\sigma|} x_a y_a \in \mathbb{R}.$$

**Types :**

$[[\sigma]]$  is a finiteness space, that is  $\begin{cases} [[\sigma]] \subseteq \mathbb{R}^{|\sigma|} \\ [[\sigma]]^{\perp\perp} = [[\sigma]] \end{cases}$

with  $[[\sigma]]^\perp = \{x \in \mathbb{R}^{|\sigma|} \mid \forall y \in [[\sigma]], x \perp y\}$

## Properties :

- $\llbracket \sigma \rrbracket$  is a linear space with infinite dimension
- $\llbracket \sigma \rrbracket$  is endowed with a linearized topology
- opens and bounded are orthogonal

## Linear Program :

$$P : \sigma \multimap \tau.$$

$\llbracket P \rrbracket : \llbracket \sigma \rrbracket \rightarrow \llbracket \tau \rrbracket$  is a continuous linear map.

## Usual Program :

$$P : \sigma \Rightarrow \tau.$$

$\llbracket P \rrbracket : \llbracket \sigma \rrbracket \rightarrow \llbracket \tau \rrbracket$  is an analytic function :

$$\forall x \in \llbracket \sigma \rrbracket, \quad P(x) = \sum_{k \leq n} P_k(\underbrace{x, \dots, x}_k) \quad \text{with } P_k \text{ the } k^{\text{th}} \text{ linearization of } P;$$

## Proposition :

Finiteness Spaces interpret differential  $\lambda$ -calculus but no fixpoints.

**Types :**  $[[\sigma]]$  is a **convenient vector space**

- Locally Convex Vector Spaces over  $\mathbb{R}$  (usual).
- Duality bounded vs. opens
- Mackey complete

**Linear Programs :**  $P : \sigma \multimap \tau$

$[[P]] \in \mathcal{L}_c([[ \sigma ]], [[ \tau ]])$  is linear and continuous.

**Usual Programs :**  $P : \sigma \Rightarrow \tau$

$[[P]] \in \mathcal{C}^\infty([[ \sigma ]], [[ \tau ]])$  is smooth.

i.e. preserves smooth curves.

**Proposition :** Convenient Vector Spaces interpret differential  $\lambda$ -calculus **without** reference to basis and with **usual** topology.

## Semantics vs Syntax :

### The full abstraction question

« *Decide what you want to say before you worry how you are going to say it.* »      The Scott-Strachey Approach to Programming Language Theory, preface, Scott (77)

## Denotational semantics :

a program as a function between mathematical spaces

## Operational semantics :

a program as a sequence of computation steps

« *Full Abstraction studies connections between denotational and operational semantics.* »      LCF Considered as a Programming Language, Plotkin (77)

# Full Abstraction = Adequacy + Full completeness

**FA relates Semantical and Observational equivalences :**

$$\begin{array}{ccc} \llbracket M \rrbracket = \llbracket N \rrbracket & \begin{array}{c} \text{Adequacy} \\ \Rightarrow \\ \Leftarrow \\ \text{Full Completeness} \end{array} & M \simeq_o N \\ & & (\forall C[\cdot], C[M] \rightarrow v \iff C[N] \rightarrow v) \end{array}$$

**How to prove Full Completeness :**

- 1 By **contradiction**, start with  $\llbracket M \rrbracket \neq \llbracket N \rrbracket$
- 2 Find **testing context** :  $f$  such that  $f \llbracket M \rrbracket \neq f \llbracket N \rrbracket$
- 3 Prove **definability** :  
 $\exists C[\cdot], \forall M, f \llbracket M \rrbracket = \llbracket C[M] \rrbracket$  and  $C[M] \rightarrow m$ .
- 4 Conclude :  
 $\exists C[\cdot], \llbracket C[M] \rrbracket \neq \llbracket C[N] \rrbracket \Rightarrow m \neq n \Rightarrow M \not\simeq_o N$ .

## Syntax vs Semantics :

### Probabilistic PCF

# A Typed Probabilistic Functional Programming Language

Integers :

$\underline{n} : \text{nat}$

$\text{pred}(\underline{k+1}) \xrightarrow{1} \underline{k}$

$\text{succ}(\underline{k}) \xrightarrow{1} \underline{k+1}$

Functions and Composition :

$(\lambda_{x:\sigma} M) N \xrightarrow{1} M[N/x]$   
 $\sigma \Rightarrow \tau$                        $\sigma$                        $\tau$

Fixpoints :

$\text{fix}(M) \xrightarrow{1} (M)\text{fix}(M)$

Case Zero :

if  $(\underline{0} = \underline{0})$  then  $P_1$  else  $P_2 \xrightarrow{1} P_1$                       + Context Rules

if  $(\underline{k+1} = \underline{0})$  then  $P_1$  else  $P_2 \xrightarrow{1} P_2$

Probabilities : for  $p + q \leq 1$

$p \cdot M + q \cdot N \xrightarrow{p} M$

$p \cdot M + q \cdot N \xrightarrow{q} N$

where  $M \xrightarrow{\rho} M'$  means that :  
 $M$  reduces to  $M'$  with probability  $\rho$



# Probabilistic Coherent Spaces

## Definition and Adequacy

**Types :**

$$\boxed{\llbracket \sigma \rrbracket \subseteq (\mathbb{R}^+)^{|\sigma|}}$$

**Example :**

$|\text{nat}|$  is the set  $\mathbb{N}$  of natural numbers

$\llbracket \text{nat} \rrbracket$  is the set of subprobability distributions over  $\mathbb{N}$ .

**Programs :**

$$\boxed{\text{For } M : \sigma, \llbracket M \rrbracket \in \llbracket \sigma \rrbracket}$$

**Example :**

$\frac{1}{2} \cdot \underline{n} + \frac{1}{3} \cdot \underline{m}$  is interpreted by  $(0, \dots, 0, \frac{1}{2}, 0, \dots, 0, \frac{1}{3}, 0, \dots)$

**Adequacy Lemma :**

Let  $M : \text{nat}$  be a closed program. Then for all  $n$ ,

$$\text{Proba}(M \xrightarrow{*} n) = \llbracket M \rrbracket_n.$$

**Types :**

$$\llbracket \sigma \Rightarrow \tau \rrbracket \subseteq (\mathbb{R}^+)^{\mathcal{M}_{\text{fin}}(|\sigma|) \times |\tau|}$$

**Example :**

$\llbracket \text{nat} \Rightarrow \text{nat} \rrbracket$  set of functions preserving subprobability distributions.

**Programs :** For  $M : \sigma \Rightarrow \tau$ ,  $\llbracket M \rrbracket : (\mathbb{R}^+)^{|\sigma|} \rightarrow (\mathbb{R}^+)^{|\tau|}$

$$\llbracket M \rrbracket(x)_- = \sum_{\mu \in \mathcal{M}_{\text{fin}}(|\sigma|)} \llbracket M \rrbracket_{\mu, -} \cdot x^\mu$$

- $x \in (\mathbb{R}^+)^{|\sigma|}$
- $\llbracket M \rrbracket_{\mu, -}$  coefficients
- $x^\mu = \prod_{a \in \text{Supp}(x)} x_a^{\mu(a)}$

**Compositionality :**

For  $P : \sigma \Rightarrow \tau$ ,  $M : \sigma$ ,

$$\llbracket (P)M \rrbracket_- = \sum_{\mu \in \mathcal{M}_{\text{fin}}(|\sigma|)} \llbracket P \rrbracket_{\mu, -} \llbracket M \rrbracket^\mu$$

## Probabilistic Full Abstraction :

### The completeness theorem

**FA relates Semantical and Observational equivalences :**

Let  $M, N : \sigma$   $\quad \forall \alpha \in |\sigma|, \llbracket M \rrbracket_\alpha = \llbracket N \rrbracket_\alpha$

Adequacy  $\Downarrow \Uparrow$  Full Completeness

$\forall P : \sigma \Rightarrow \text{nat}, \forall n \in |\text{nat}|,$   
 $\text{Proba}((P)M \xrightarrow{*} n) = \text{Proba}((P)N \xrightarrow{*} n)$

**How to prove Adequacy :**

- ① Apply Adequacy Lemma :

$$\forall n, \text{Proba}((P)M \xrightarrow{*} n) = \llbracket (P)M \rrbracket_n.$$

- ② Apply Compositionality :

$$\forall n, \llbracket (P)M \rrbracket_n = \sum_{\mu \in \mathcal{M}_{\text{fin}}(|\sigma|)} \llbracket P \rrbracket_{\mu, n} \prod_{\alpha \in \mu} \llbracket M \rrbracket_\alpha^{\mu(\alpha)}$$

## FA relates Semantical and Observational equivalences :

Let  $M, N : \sigma$   $\quad \forall \alpha \in |\sigma|, \llbracket M \rrbracket_\alpha = \llbracket N \rrbracket_\alpha$

Adequacy  $\Downarrow \Uparrow$  Full Completeness

$\forall P : \sigma \Rightarrow \text{nat}, \forall n \in |\text{nat}|,$   
 $\text{Proba}((P)M \xrightarrow{*} n) = \text{Proba}((P)N \xrightarrow{*} n)$

## How to prove Full Completeness :

- ① By **contradiction** :  $\exists \alpha \in |\sigma|, \llbracket M \rrbracket_\alpha \neq \llbracket N \rrbracket_\alpha$
- ② Find **testing context** :  $P_\alpha$  such that  $\llbracket (P_\alpha)M \rrbracket_0 \neq \llbracket (P_\alpha)N \rrbracket_0$
- ③ Prove **definability** :  $P_\alpha \in \text{PPCF}$
- ④ Apply **Adequacy** :  $\text{Proba}((P_\alpha)M \xrightarrow{*} 0) \neq \text{Proba}((P_\alpha)N \xrightarrow{*} 0)$ .

# How to prove Full Completeness :

- 1 By **contradiction** :  $\exists \alpha \in |\sigma|$ ,  $\llbracket M \rrbracket_\alpha \neq \llbracket N \rrbracket_\alpha$
- 2 Find **testing context** :  $P_\alpha$  such that  $\llbracket (P_\alpha)M \rrbracket_0 \neq \llbracket (P_\alpha)N \rrbracket_0$ 
  - **Base case** :  $\sigma = \text{nat}$ ,  $\alpha = n$ , take

If  $P_n = \lambda x^t$  if  $(x = \underline{n})$  then  $\underline{0}$       Then       $\llbracket (P_n)M \rrbracket_0 = \llbracket M \rrbracket_n$

- **Induction case** : by Compositionality,

$$\llbracket (P_\alpha(\vec{X}))M \rrbracket_0 = \sum_{\mu \in \mathcal{M}_{\text{fin}}(|\sigma|)} \llbracket P_\alpha(\vec{X}) \rrbracket_{\mu,0} \prod_{\delta \in \mu} \llbracket M \rrbracket_\delta^{\mu(\delta)}$$

If  $\llbracket P_\alpha(\vec{X}) \rrbracket_{\mu,0}$  is

- a power series in  $\vec{X}$
- with coeff of  $\prod \vec{X} \neq 0 \iff \mu = [\alpha]$

Then  $\llbracket (P_\alpha(\vec{X}))M \rrbracket_0$  is

- a power series in  $\vec{X}$
- with coeff of  $\prod \vec{X}$  proportional to  $\llbracket M \rrbracket_\alpha$ .

$\llbracket (P_\alpha(\vec{X}))M \rrbracket_0$  and  $\llbracket (P_\alpha(\vec{X}))N \rrbracket_0$  are different power series

- ② Find **testing context** :  $\forall \vec{\lambda} \in [0, 1]^{\mathbb{N}}, P_{\alpha}(\vec{\lambda}) \in PPCF$  and the series  $[(P_{\alpha}(\vec{\lambda}))M]_0$  and  $[(P_{\alpha}(\vec{\lambda}))N]_0$  converge absolutely with **different coefficients**.
- ③ Prove **definability** :

$$\exists \vec{\lambda} \in [0, 1]^{\mathbb{N}}, [(P_{\alpha}(\vec{\lambda}))M]_0 \neq [(P_{\alpha}(\vec{\lambda}))N]_0.$$

By contradiction :

- If they were equal, their derivatives near zero would be equal.
- Coefficients of power series are computed by **derivation** at zero.





⚠ PCoh is NOT a model of differential lambda-calculus.

- ④ Apply **Adequacy** :  $\exists \vec{\lambda} \in [0, 1]^{\mathbb{N}}, P_{\alpha}(\vec{\lambda}) \in PPCF$

$$\text{Proba}((P_{\alpha}(\vec{\lambda}))M \xrightarrow{*} 0) \neq \text{Proba}((P_{\alpha}(\vec{\lambda}))N \xrightarrow{*} 0).$$



- 1 **Syntactical** Taylor expansion and Resource consumption
- 2 Taylor expansion in **Semantics**
- 3 **Semantics vs Syntax** : The full abstraction question

-  [Ehrhard, Regnier] *Uniformity and the Taylor expansion of ordinary lambda-terms*, 2008
-  [Girard], *Between logic and quantic : a tract*, 1999
-  [Ehrhard], *Finiteness Spaces*, 2005
-  [Kriegle, Michor], *The Convenient Setting of Global Analysis*, 1997